

FU

# BREVET D'INVENTION

**CERTIFICAT D'UTILITÉ - CERTIFICAT D'ADDITION****097857732**

## COPIE OFFICIELLE

Le Directeur général de l'Institut national de la propriété industrielle certifie que le document ci-annexé est la copie certifiée conforme d'une demande de titre de propriété industrielle déposée à l'Institut.

Fait à Paris, le **24 JAN. 2000**

Pour le Directeur général de l'Institut  
national de la propriété industrielle  
Le Chef du Département des brevets

### DOCUMENT DE PRIORITÉ

PRÉSENTÉ OU TRANSMIS  
CONFORMÉMENT À LA REGLE  
17.1.a) OU b)

Martine PLANCHE

INSTITUT  
NATIONAL DE  
LA PROPRIÉTÉ  
INDUSTRIELLE

SIEGE  
26 bis, rue de Saint Petersburg  
75800 PARIS Cédex 08  
Téléphone : 01 53 04 53 04  
Télécopie : 01 42 93 59 30

This Page Blank (uspto)

**REQUÊTE EN DÉLIVRANCE**

26 bis, rue de Saint Pétersbourg  
75800 Paris Cedex 08  
Téléphone : 01 53 04 53 04 Télécopie : 01 42 93 59 30

Confirmation d'un dépôt par télécopie ☐

Cet imprimé est à remplir à l'encre noire en lettres capitales

Réservé à l'INPI

DATE DE REMISE DES PIÈCES

N° D'ENREGISTREMENT NATIONAL

DÉPARTEMENT DE DÉPÔT

DATE DE DÉPÔT

08.12.98  
98 154 93 -

75

08 DEC. 1998

1 NOM ET ADRESSE DU DEMANDEUR OU DU MANDATAIRE  
À QUI LA CORRESPONDANCE DOIT ÊTRE ADRESSÉE

**SCHLUMBERGER SYSTEMES**

50 Av. Jean Jaurès - B.P 620-04  
92542 MONTROUGE CEDEX

A l'attention de Anne DANG TRAN

n° du pouvoir permanent PG 73 91  
références du correspondant 76-0552  
téléphone 01 47 46 72 14

2 DEMANDE Nature du titre de propriété industrielle

☒ brevet d'invention

☐ demande divisionnaire

☐ certificat d'utilité

☐ transformation d'une demande  
de brevet européen

☐ demande initiale

☐ brevet d'invention

☐ certificat d'utilité n°

date

Établissement du rapport de recherche

☐ différé

☒ immédiat

Le demandeur, personne physique, requiert le paiement échelonné de la redevance

☐ oui

☐ non

Titre de l'invention (200 caractères maximum)

**DISPOSITIF ET PROCEDE D'INITIALISATION D'UN PROGRAMME APPLICATIF  
D'UNE CARTE A CIRCUIT INTEGRE**

3 DEMANDEUR (S)

n° SIREN 5 6 2 1 1 3 5 3 0

code APE-NAF

Nom et prénoms (souligner le nom patronymique) ou dénomination

**Schlumberger Systèmes**

Forme juridique

**Société Anonyme**

Nationalité (s)

**Française**

Adresse (s) complète (s)

Pays

**50, Avenue Jean Jaurès  
92120 MONTROUGE**

**France**

En cas d'insuffisance de place, poursuivre sur papier libre ☐

4 INVENTEUR (S) Les inventeurs sont les demandeurs

☐ oui

☒ non

Si la réponse est non, fournir une désignation séparée

5 RÉDUCTION DU TAUX DES REDEVANCES

☐ requise pour la 1ère fois

☐ requise antérieurement au dépôt ; joindre copie de la décision d'admission

6 DÉCLARATION DE PRIORITÉ OU REQUÊTE DU BÉNÉFICE DE LA DATE DE DÉPÔT D'UNE DEMANDE ANTÉRIEURE

pays d'origine

numéro

date de dépôt

nature de la demande

**SANS**

7 DIVISIONS antérieures à la présente demande

n°

date

n°

date

8 SIGNATURE DU DEMANDEUR OU DU MANDATAIRE

(nom et qualité du signataire - n° d'inscription)

**Anne DANG TRAN**

**Mandataire (PG 73 91)**

SIGNATURE DU PRÉPOSÉ À LA RÉCEPTION

SIGNATURE APRÈS ENREGISTREMENT DE LA DEMANDE À L'INPI



# BREVET D'INVENTION, CERTIFICAT D'UTILITE

DÉSIGNATION DE L'INVENTEUR  
(si le demandeur n'est pas l'inventeur ou l'unique inventeur)

DEPARTEMENT DES BREVETS

26bis, rue de Saint-Petersbourg  
75800 Paris Cédex 08  
Tél. : 01 53 04 53 04 - Télécopie : 01 42 93 59 30

76-0552

N° D'ENREGISTREMENT NATIONAL

9815493

TITRE DE L'INVENTION :

DISPOSITIF ET PROCEDE D'INITIALISATION D'UN PROGRAMME APPLICATIF  
D'UNE CARTE A CIRCUIT INTEGRE

LE(S) SOUSSIGNÉ(S)

**Anne DANG TRAN**  
**SCHLUMBERGER SYSTEMES**  
50, avenue Jean Jaurès - BP 620-04  
92542 MONTROUGE CEDEX

DÉSIGNE(NT) EN TANT QU'INVENTEUR(S) (indiquer nom, prénoms, adresse et souligner le nom patronymique) :

**BURIANNE Yannick**  
1 Allée des noisetiers  
92140 CLAMART

NOTA : A titre exceptionnel, le nom de l'inventeur peut être suivi de celui de la société à laquelle il appartient (société d'appartenance) lorsque celle-ci est différente de la société déposante ou titulaire.

Date et signature (s) du (des) demandeur (s) ou du mandataire

Le 8 décembre 1998

**Anne DANG TRAN**  
**(PG 73 91)**

# DOCUMENT COMPORTANT DES MODIFICATIONS

PAGE(S) DE LA DESCRIPTION OU DES REVENDECATIONS OU PLANCHE(S) DE DESSIN			R.M.*	DATE DE LA CORRESPONDANCE	TAMPON DATEUR DU CORRECTEUR
Modifiée(s)	Supprimée(s)	Ajoutée(s)			
1				20 MAI 99	31 MAI 1999 A J P
9, 10			x	20 MAI 99	31 MAI 1999 A J P

Un changement apporté à la rédaction des revendications d'origine, sauf si celui-ci découle des dispositions de l'article R.612-36 du code de la Propriété Intellectuelle, est signalé par la mention «R.M.» (revendications modifiées).

## **DISPOSITIF ET PROCEDE D'INITIALISATION D'UN PROGRAMME APPLICATIF D'UNE CARTE A CIRCUIT INTEGRE**

La présente invention concerne un dispositif à circuit intégré comprenant une mémoire et au moins un programme applicatif résident dans ladite mémoire. Elle concerne également un procédé d'initialisation d'un programme applicatif d'un tel dispositif.

5 Lesdits dispositifs sont en particulier des objets portatifs appelés cartes à puce comprenant des programmes applicatifs concernant le domaine de la santé, de la téléphonie mobile, ou encore, concernant le domaine bancaire.

10 Lesdites cartes à puce comportent un corps de carte dans lequel est intégré un module électronique contenant de manière classique un élément de commande (par exemple une unité centrale de traitement ou CPU) et une mémoire. Ladite mémoire comporte au moins un programme applicatif contenant des éléments unitaires auxquels on affecte des valeurs afin que le programme puisse être exécuté, lesdits  
15 éléments n'étant pas modifiés lors de l'exécution dudit programme applicatif. Ces éléments sont appelés variables configurables.

En vue de configurer lesdites variables, l'état de la technique propose des dispositifs qui prévoient des fichiers contenant des données qui sont affectées aux variables lors d'une phase dite d'initialisation.  
20 Cette phase d'initialisation est nécessaire au bon déroulement du programme applicatif. A cet effet, lesdits dispositifs comportent un moyen de commande qui permet de modifier les valeurs desdites données d'initialisation dans lesdits fichiers et ensuite d'affecter ces données auxdites variables. Lorsque ces variables sont stockées en  
25 mémoire de façon permanente, elles conservent leur valeur d'initialisation même si la carte n'est plus alimentée en tension.

Bien que ces dispositifs permettent de configurer un programme applicatif, les valeurs d'initialisation sont dupliquées dans deux espaces

mémoire de tailles quasi identiques, l'un contenant les fichiers de données d'initialisation et l'autre étant l'espace alloué pour les variables qui sont initialisées avec lesdites données, ce qui peut être gênant du fait de la taille limitée de la mémoire des cartes à puce. De plus, le  
5 temps d'exécution dudit programme applicatif est sensiblement accru du fait notamment de la nécessité d'effectuer ladite phase d'initialisation lors de chaque exécution du programme même si les valeurs d'initialisation n'ont pas changées car ladite phase d'initialisation fait partie intégrante du programme applicatif. Enfin, il  
10 existe des cas où, soit le programme applicatif ne possède aucun privilège pour accéder auxdits fichiers, soit ladite carte ne possède tout simplement aucun fichier.

Aussi un problème technique à résoudre par l'objet de la présente invention est de proposer un dispositif à circuit intégré comprenant une  
15 mémoire et au moins un programme applicatif résident dans ladite mémoire, ainsi qu'un procédé d'initialisation d'un programme applicatif d'un tel dispositif, qui permettraient, d'une part, de configurer un programme applicatif sans avoir de duplication de données et ainsi éviter des pertes d'espace mémoire dues aux fichiers précités, et, d'autre  
20 part, d'éviter d'augmenter le temps d'exécution dudit programme applicatif.

Une solution au problème technique posé se caractérise, selon un premier objet de la présente invention, en ce que ledit programme applicatif comprend au moins une variable configurable et une liste  
25 d'au moins un élément référence, et en ce que ladite mémoire comporte, d'une part, au moins un moyen d'initialisation desdites variables, ledit moyen étant paramétré par plusieurs paramètres dont l'un des paramètres est ladite liste d'éléments références, et, d'autre part, une commande permettant d'envoyer des données contenant en particulier  
30 des valeurs à affecter aux variables configurables.

Selon un second objet de la présente invention, cette solution se caractérise en ce que le procédé d'initialisation comporte les étapes consistant à :

- créer, dans ledit programme applicatif, au moins une variable configurable et une liste d'au moins un élément référence,
- envoyer des données contenant en particulier des valeurs à affecter aux variables configurables,
- initialiser lesdites variables grâce à un moyen d'initialisation, ledit moyen étant paramétré par plusieurs paramètres dont l'un des paramètres est ladite liste d'éléments références.

Ainsi, comme on le verra en détail plus loin, le dispositif de l'invention permet d'avoir une gestion optimisée de la mémoire de la carte et une configuration directe des variables d'un programme applicatif grâce à la commande qui permet de modifier les valeurs affectées aux variables configurables et grâce également à la liste d'éléments références passée en paramètre du moyen d'initialisation, liste qui permet d'établir un lien entre les valeurs envoyées par ladite commande et les variables du programme applicatif à configurer.

La description qui va suivre au regard des dessins annexés, donnée à titre d'exemple non limitatif, fera bien comprendre en quoi consiste l'invention et comment elle peut être réalisée.

La figure 1 est un schéma d'un dispositif à circuit intégré, ici une carte à puce.

La figure 2 est un schéma d'une mémoire de la carte de la figure 1.

La figure 3 est un schéma d'un programme applicatif de la carte de la figure 1.

La figure 4 est un schéma d'une commande de la carte de la figure 1.



La figure 5 est un schéma d'une liste d'éléments d'un programme applicatif de la mémoire de la figure 2.

La figure 6 est un autre schéma de la mémoire de la carte de la figure 1.

5 La figure 7 est un schéma montrant des variables contenues dans le programme applicatif de la figure 3.

Sur la figure 1 est représenté un dispositif 10 à circuit intégré, ici une carte à puce.

Cette carte 10 contient un élément 11 de commande (par exemple  
10 une unité centrale de traitement ou CPU), une mémoire 12 et un bloc 13 de contacts destiné à une connexion électrique avec par exemple un connecteur d'un lecteur de cartes.

Ladite mémoire 12 est représentée sur la figure 2. Elle comprend un programme applicatif A. Ledit programme A comprend au moins une  
15 variable configurable V et une liste L d'au moins un élément référence R. Ladite mémoire comporte, d'une part, au moins un moyen MI d'initialisation desdites variables V, ledit moyen étant paramétré par plusieurs paramètres dont l'un des paramètres est ladite liste L d'éléments références, et, d'autre part, une commande CDE permettant  
20 d'envoyer des données contenant en particulier des valeurs à affecter aux variables configurables. Le moyen MI est une fonction ou un bout de programme. Sur la figure 3, le programme applicatif A comporte trois variables configurables V1, V2 et V3 et une liste L contenant trois éléments références R1, R2 et R3.

25 Afin que le programme A se déroule correctement, il faut configurer ses variables, c'est à dire leur affecter des valeurs.

Dans une première étape, la commande CDE est envoyée à la carte 10. Elle comporte des données telles que par exemple, un nombre d'éléments références R, des numéros indexant les éléments références  
30 d'une liste, des valeurs associées.... Sur la figure 4, la commande CDE

envoie les trois valeurs alphanumériques suivantes APPLICATION GSM, TELEPHONER et APPEL EN COURS. Ces valeurs sont précédées des index 1, 2 et 3 qui correspondent à trois éléments références.

Lorsque le programme applicatif A reçoit la commande CDE, il est exécuté et la phase d'initialisation faisant appel au moyen MI commence.

Dans une deuxième étape, on construit un lien entre les valeurs envoyées par la commande CDE et les éléments références d'une liste L spécifique. La liste L d'éléments références paramétrant le moyen MI d'initialisation permet d'établir ce lien. Les autres paramètres sont entre autres les données envoyées par ladite commande CDE. On spécifie la liste L en donnant par exemple son nom. Sur la figure 5, L est appelée CUSTOMELEMENT. Elle contient trois éléments références MENU, TEXT et MESSAGE auxquels sont associées les valeurs alphanumériques respectives APPLICATION GSM, TELEPHONER et APPEL EN COURS. Ces valeurs proviennent de la commande CDE.

Dans une troisième étape, le moyen MI d'initialisation établit un lien entre lesdites valeurs de ladite liste L et les variables à configurer V grâce aux éléments références R. A cet effet, un élément référence R fait référence à une variable configurable V. Sur la figure 3, R1, R2 et R3 font respectivement référence aux variables V1, V2 et V3, ces dernières étant des variables dont on veut initialiser tout ou partie de leur contenu. C'est grâce à ces différents liens que le transfert des valeurs vers lesdites variables s'effectue.

Une fois ce transfert effectué, la configuration du programme applicatif A est terminée et la suite dudit programme peut se dérouler comme souhaité. Le dispositif selon l'invention ne comporte aucun fichier, de ce fait, la configuration des variables a été directe.

On notera que l'invention prévoit également que ladite commande CDE permet de lire le contenu des variables configurables et ce grâce à

la présence d'un paramètre de ladite commande appelé MODE qui indique si la commande doit envoyer ou lire des données. Ceci permet de lire les valeurs des variables V à tout moment et par conséquent de connaître la configuration du programme applicatif A à tout moment.

5 Il peut être intéressant pour gagner de l'espace mémoire et homogénéiser les initialisations de permettre à un ou plusieurs programmes applicatifs d'utiliser le même moyen MI d'initialisation. Aussi, au moins un moyen MI d'initialisation réside dans ladite mémoire indépendamment d'un programme applicatif A. Cela signifie  
10 que ledit moyen MI peut être utilisé par tous les programmes applicatifs résidents dans la carte 10 et n'est propre à aucun programme A en particulier. Comme le montre la figure 6, le moyen MI1 est indépendant des programmes applicatifs A1 et A2 et peut être de ce fait utilisé par l'un ou l'autre de ces programmes.

15 Cependant, il peut être également utile de pouvoir personnaliser le moyen d'initialisation pour un programme applicatif donné en ayant un moyen différent de MI1, par exemple dans le cas où l'on veut avoir un protocole d'échange de données différent de celui de MI1 c'est à dire un format de données d'initialisation différent. Comme le montre la  
20 figure 6, au moins un programme applicatif A2 comprend un moyen MI2 d'initialisation. Pour configurer les variables de A2, on aura le choix d'utiliser les moyens MI1 ou MI2 si lesdites variables respectent le format de données respectif desdits moyens.

On notera que l'on peut également n'avoir aucun moyen MI  
25 indépendant d'un programme applicatif, chaque moyen MI d'initialisation étant, dans ce cas, propre à un programme applicatif, ou au contraire n'avoir que des moyens indépendants.

La présente invention s'applique particulièrement à des programmes applicatifs qui sont programmés dans des langages de  
30 haut niveau tels qu'en particulier un langage appelé JAVA (marque

déposée). Ce langage traite des notions de classe, d'héritage, d'attribut et de méthode bien connues de l'homme du métier.

Dans le cas où le programme applicatif A est programmé en JAVA, les variables configurables sont des objets et une liste d'éléments 5 références fait référence à un ensemble d'objets. Sur la figure 7, la mémoire 12 comprend un programme applicatif A. Ledit programme applicatif A comporte au moins deux variables V1 et V3 configurables référencées dans une même liste et qui dérivent d'une même classe mère C0. De plus, ledit programme applicatif A comporte au moins 10 deux variables V1 et V2 configurables référencées dans une même liste et qui sont des instances d'une même classe C1. Les différentes classes sont définies soit dans le programme applicatif A, soit de façon indépendante, par exemple dans une librairie. Lesdites variables configurables sont persistantes dans ladite mémoire 12.

15 On peut voir que ladite liste L représente des objets ayant, soit des points en commun, les variables ou objets V1 et V3 héritent de l'attribut At1 et des méthodes M1 et M2 de la classe C0 mais ont leur propres attributs et méthodes, soit tous leurs points en commun, V1 et V2 sont des instances de la classe C1 qui possède l'attribut At2 et la 20 méthode M3. Pour configurer lesdits objets, il faut qu'une liste L soit du même type qu'une classe mère ou que la classe desdits objets. Ainsi un moyen MI1 simple permettra de configurer une partie du contenu des objets V1, V2 et V3, soit l'attribut At1. On pourrait également avoir un autre moyen MI2 plus complexe permettant de configurer l'ensemble 25 des attributs At1 et At2 des variables V1 et V2.

C'est grâce à la définition du type de ladite liste L que la présente invention nous permet de modifier les valeurs des attributs d'objets bien spécifiés et d'empêcher ainsi la modification par inadvertance du contenu d'autres objets. De plus, grâce à la présente invention, il n'y a 30 pas d'accès direct à l'emplacement mémoire contenant toutes les

variables du programme applicatif A et, par suite, on ne risque pas de modifier de façon frauduleuse toutes ces variables.

Un autre avantage de la présente invention est que lesdites variables ou objets sont persistants en mémoire. Cela signifie qu'une fois configurés et lorsqu'ils ne sont pas modifiés pendant l'exécution du programme applicatif A, lesdits objets conservent leurs valeurs d'initialisation même après l'exécution du programme A. Si on ne veut pas modifier ces valeurs avant une autre exécution de A, il est inutile pour un utilisateur d'envoyer la commande CDE pour reconfigurer le programme applicatif A. Par suite, on s'affranchit de la phase d'initialisation et aucun moyen MI d'initialisation n'est déclenché. Par conséquent, le temps d'exécution est diminué.

Comme nous venons de le voir, le langage JAVA est intéressant à plus d'un égard, mais une de ses caractéristiques qui fait également sa force est qu'il possède des moyens sécuritaires dont un moyen qui vérifie que chaque instruction d'un programme applicatif A est valide ainsi que les paramètres de cette instruction. Par exemple, si une instruction nécessite un tableau d'octets situé à une certaine adresse de la mémoire 12 comme paramètre alors qu'une adresse mémoire interdite est désignée à la place, ledit moyen sécuritaire permettra de détecter cette erreur et d'empêcher ainsi l'accès à un espace mémoire interdit. Afin de profiter de ces moyens sécuritaires de vérification, l'invention prévoit que tout moyen MI d'initialisation est défini dans le même langage que ledit programme applicatif A, c'est à dire en JAVA. Ainsi, si un paramètre dudit moyen MI est faux, le programme ne sera pas exécuté et un fraudeur ne pourra accéder à des emplacements mémoire interdits.

### **REVENDEICATIONS**

- 5       **1** - Dispositif à circuit intégré comprenant une mémoire et au moins un programme applicatif résident dans ladite mémoire, caractérisé en ce que ledit programme applicatif comprend au moins une variable configurable et une liste d'au moins un élément référence, et en ce que ladite mémoire comporte, d'une part, au moins un moyen d'initialisation desdites variables, ledit moyen étant paramétré par plusieurs paramètres dont l'un des
- 10       paramètres est ladite liste d'éléments références, et, d'autre part, une commande permettant d'envoyer des données contenant en particulier des valeurs à affecter aux variables configurables.
- 15       **2** - Dispositif selon la revendication 1, caractérisé en ce que lesdites variables configurables sont persistantes dans ladite mémoire.
- 3** - Dispositif selon l'une des revendications précédentes, caractérisé en ce qu'un élément référence fait référence à une variable configurable.
- 20       **4** - Dispositif selon l'une des revendications précédentes, caractérisé en ce que ledit programme applicatif comporte au moins deux variables configurables référencées dans une même liste et qui dérivent d'une même classe mère.
- 25       **5** - Dispositif selon l'une des revendications précédentes, caractérisé en ce que ledit programme applicatif comporte au moins deux variables configurables référencées dans une même liste et qui sont des instances d'une même classe.
- 6** - Dispositif selon l'une des revendications précédentes, caractérisé en ce qu'au moins un moyen d'initialisation réside dans ladite mémoire indépendamment d'un programme applicatif.

**8** - Dispositif selon l'une des revendications précédentes, caractérisé en ce qu'au moins un programme applicatif comprend un moyen d'initialisation.

5 **9** - Dispositif selon l'une des revendications précédentes, caractérisé en ce que tout moyen d'initialisation est défini dans le même langage que ledit programme applicatif.

**10** - Dispositif selon l'une des revendications précédentes, caractérisé en ce que ladite commande permet de lire le contenu des variables configurables.

10 **11** - Procédé d'initialisation d'un programme applicatif d'un dispositif à circuit intégré comprenant une mémoire et au moins un programme applicatif résident dans ladite mémoire, caractérisé en ce que ledit procédé comporte les étapes consistant à :

15 -créer, dans ledit programme applicatif, au moins une variable configurable et une liste d'au moins un élément référence,  
-envoyer des données contenant en particulier des valeurs à affecter aux variables configurables,  
-initialiser lesdites variables grâce à un moyen d'initialisation,  
20 ledit moyen étant paramétré par plusieurs paramètres dont l'un des paramètres est ladite liste d'éléments références.

## **DISPOSITIF ET PROCEDE D'INITIALISATION D'UN PROGRAMME APPLICATIF D'UNE CARTE A CIRCUIT INTEGRE**

La présente invention concerne un dispositif à circuit intégré comprenant une mémoire et au moins un programme applicatif résident dans ladite mémoire. Elle concerne également un procédé d'initialisation d'un programme applicatif d'un tel dispositif.

5 Lesdits dispositifs sont en particulier des objets portatifs appelés cartes à puce comprenant des programmes applicatifs concernant le domaine de la santé, de la téléphonie mobile, ou encore, concernant le domaine bancaire.

Lesdites cartes à puce comportent un corps de carte dans lequel  
10 est intégré un module électronique contenant de manière classique un élément de commande (par exemple une unité centrale de traitement ou CPU) et une mémoire. Ladite mémoire comporte au moins un programme applicatif contenant des éléments unitaires auxquels on affecte des valeurs afin que le programme puisse être exécuté, lesdits  
15 éléments n'étant pas modifiés lors de l'exécution dudit programme applicatif. Ces éléments sont appelés variables configurables.

En vue de configurer lesdites variables, l'état de la technique propose des dispositifs qui prévoient des fichiers contenant des données qui sont affectées aux variables lors d'une phase dite d'initialisation.  
20 Cette phase d'initialisation est nécessaire au bon déroulement du programme applicatif. A cet effet, lesdits dispositifs comportent un moyen de commande qui permet de modifier les valeurs desdites données d'initialisation dans lesdits fichiers et ensuite d'affecter ces données auxdites variables. Lorsque ces variables sont stockées en  
25 mémoire de façon permanente, elles conservent leur valeur d'initialisation même si la carte n'est plus alimentée en tension.

Bien que ces dispositifs permettent de configurer un programme applicatif, les valeurs d'initialisation sont dupliquées dans deux espaces



**REVENDEICATIONS**

- 1 - Dispositif à circuit intégré comprenant une mémoire et au moins un programme applicatif résident dans ladite mémoire, caractérisé en ce que ledit programme applicatif comprend au moins une variable configurable et une liste d'au moins un élément référence, et en ce que ladite mémoire comporte, d'une part, au moins un moyen d'initialisation desdites variables, ledit moyen étant paramétré par plusieurs paramètres dont l'un des paramètres est ladite liste d'éléments références, et, d'autre part, une commande permettant d'envoyer des données contenant en particulier des valeurs à affecter aux variables configurables.
- 2 - Dispositif selon la revendication 1, caractérisé en ce que lesdites variables configurables sont persistantes dans ladite mémoire.
- 3 - Dispositif selon l'une des revendications précédentes, caractérisé en ce qu'un élément référence fait référence à une variable configurable.
- 4 - Dispositif selon l'une des revendications précédentes, caractérisé en ce que ledit programme applicatif comporte au moins deux variables configurables référencées dans une même liste et qui dérivent d'une même classe mère.
- 5 - Dispositif selon l'une des revendications précédentes, caractérisé en ce que ledit programme applicatif comporte au moins deux variables configurables référencées dans une même liste et qui sont des instances d'une même classe.
- 6 - Dispositif selon l'une des revendications précédentes, caractérisé en ce qu'au moins un moyen d'initialisation réside dans ladite mémoire indépendamment d'un programme applicatif.

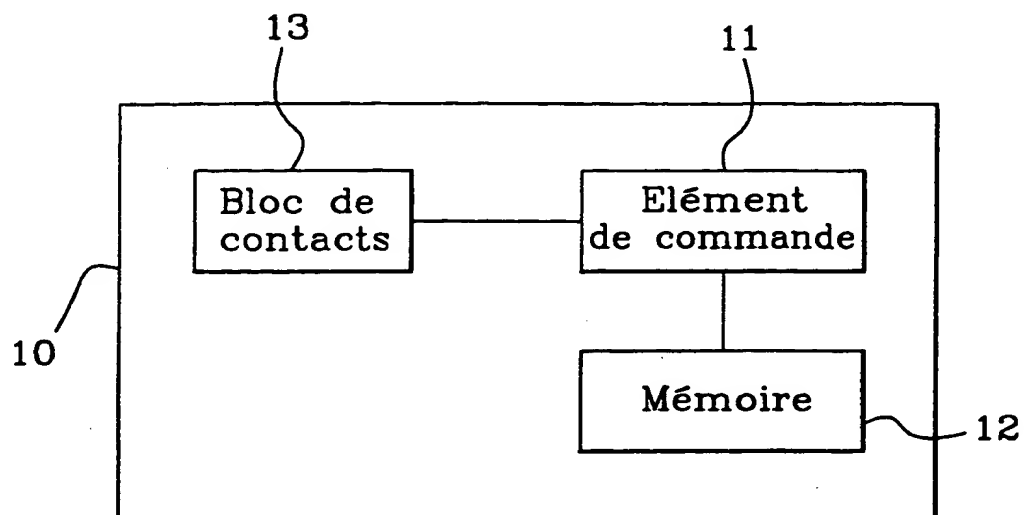
7 - Dispositif selon l'une des revendications précédentes, caractérisé en ce qu'au moins un programme applicatif comprend un moyen d'initialisation.

5 8 - Dispositif selon l'une des revendications précédentes, caractérisé en ce que tout moyen d'initialisation est défini dans le même langage que ledit programme applicatif.

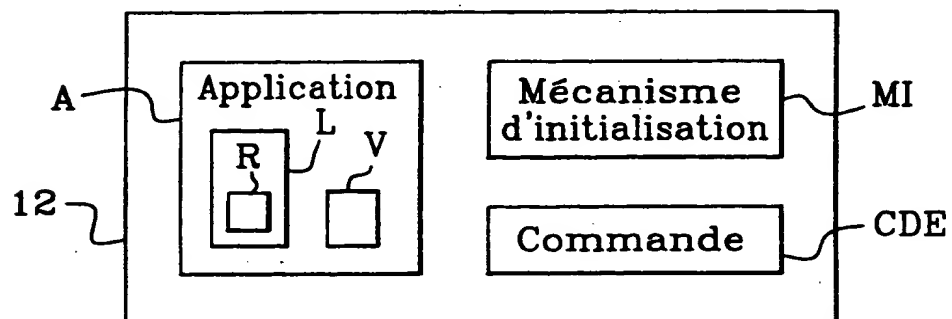
9 - Dispositif selon l'une des revendications précédentes, caractérisé en ce que ladite commande permet de lire le contenu des variables configurables.

10 10 - Procédé d'initialisation d'un programme applicatif d'un dispositif à circuit intégré comprenant une mémoire et au moins un programme applicatif résident dans ladite mémoire, caractérisé en ce que ledit procédé comporte les étapes consistant à :

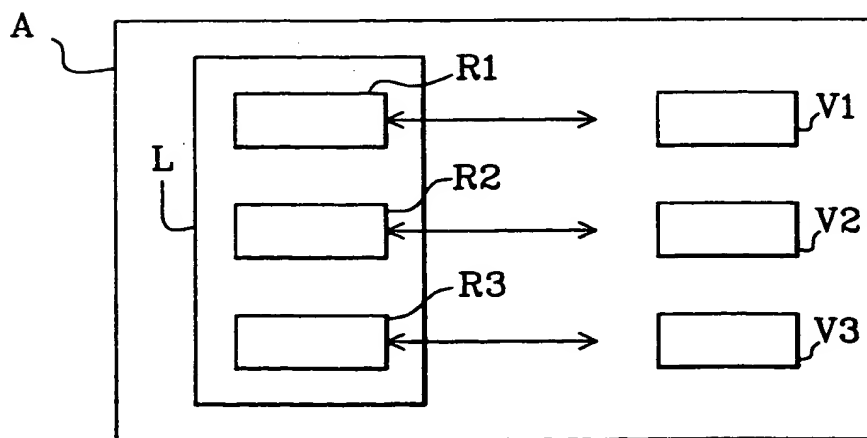
15 -créer, dans ledit programme applicatif, au moins une variable configurable et une liste d'au moins un élément référence,  
-envoyer des données contenant en particulier des valeurs à affecter aux variables configurables,  
-initialiser lesdites variables grâce à un moyen d'initialisation,  
20 ledit moyen étant paramétré par plusieurs paramètres dont l'un des paramètres est ladite liste d'éléments références.



**FIG.1**



**FIG.2**

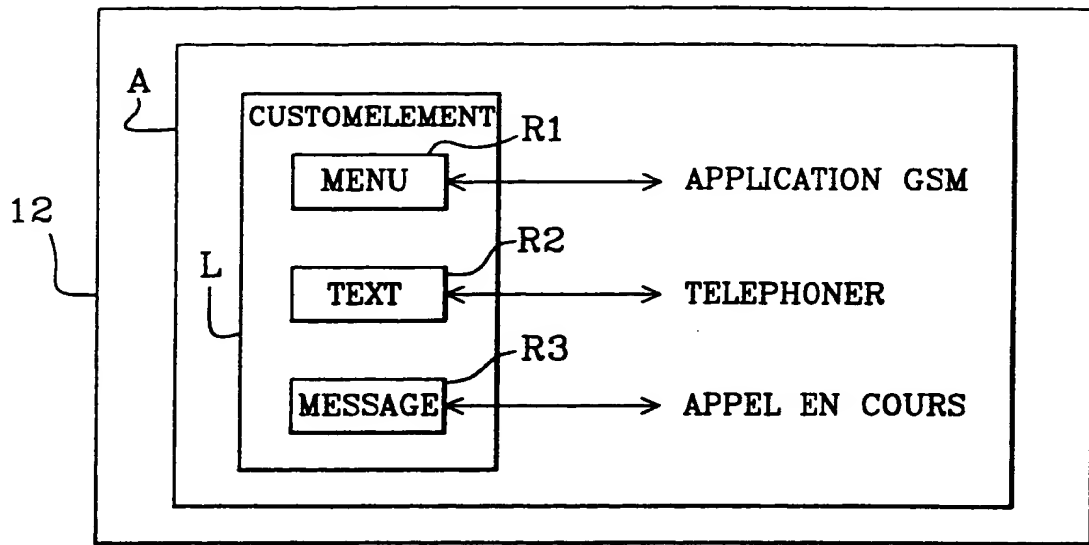


**FIG.3**

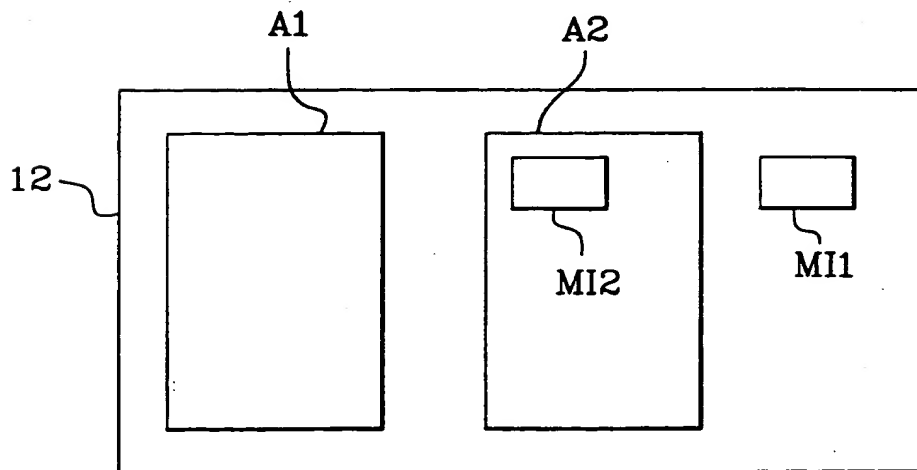
CDE	NOMBRE D'ELEMENTS MODE A CONFIGURER			LONGUEUR TOTALE DES DONNEES	INDEX	LONGUEUR DES VALEURS	VALEURS
	INS	IN					
CLASS			3	47	1	15	APPLICATION GSM
					2	10	TELEPHONER
					3	14	APPEL EN COURS

DONNEES

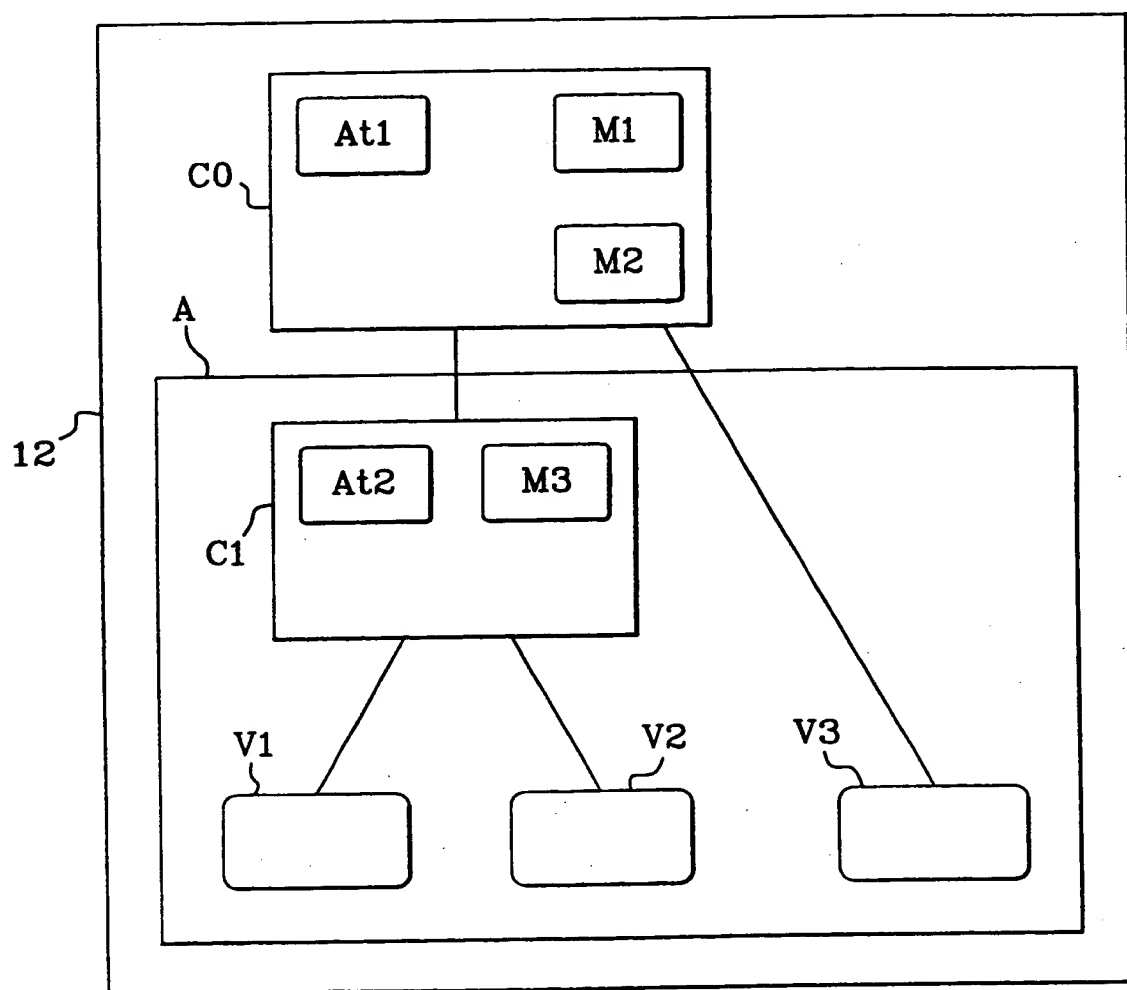
FIG.4



**FIG.5**



**FIG.6**

**FIG.7**



This Page Blank (uspto)